1

A BEHAVIOURAL TRANSLATOR FOR AN OBJECT

5 This invention relates to a behavioural translator for an object. In particular, but not exclusively, to a behavioural translator which enables the behaviour of a virtual object to be translated from behaviour associated with a first demographic group to equivalent behaviour associated with another demographic group.

10 The behavioural translator is arranged to translate the behaviour automatically generated for an object from behaviour associated with one demographic group to behaviour associated with another demographic group. A behavioural translator thus enables one observer of a virtual object participating in a virtual environment to perceive the virtual object behaving in accordance with a demographic group with may differ from the behaviour perceived by another observer participating in the same virtual environment A

15 demographic group is any group of person which has a distinctive culture, for example, a group with distinctive religious, political, social, language, racial, national, and/or other characteristics.

An example of a method of and apparatus for generating behaviour in an object for which

20 the behavioural translator may be implemented as described herein forms the subject of the inventors co-pending PCT patent application entitled "APPARATUS AND METHOD FOR GENERATING BEHAVIOUR IN AN OBJECT". Although the invention is not limited thereto, one application of particular interest is the generation of autonomous behaviour conveying body language in a virtual agent or object, for example an avatar which may

25 be translated in a virtual environment in which a plurality of human users from differing cultural backgrounds participate.

Body language in particular may differ from culture to culture. More information on body language can be obtained from "Peoplewatching: The Desmond Morris Guide to Body

30 Language", ISBN: 0099429780, the contents of which are hereby incorporated by reference.

2

Animated objects, whether virtual (such as an avatar) or robotic (for example, a pet "toy") are becoming increasingly sophisticated in their behaviour. In particular, there is a consumer demand for more realistic behaviour by virtual objects such as avatars and animated agents. In the context of this invention, an avatar is defined to be a visual

5   representation of a user in a virtual environment, taking any appropriate form. An animated agent is an intelligent software based agent used for interactions with a user in a virtual environment, or for interactive entertainment. Similarly, the animated agent may take any appropriate form.

10 When an object is capable of having animated behaviour, it is desirable for a user who is relatively unskilled in programming to have the ability to personalise the object's behaviour. This is particularly so when the user is seeking, for example, to create a sense of individuality in a group setting, to enhance a role the object is playing, or to reflect the user's own personality. To make objects more interesting to their human

15 users, it is desirable if a personality can be provided for an object, i.e., if the behaviour can be modified by the user to have certain traits that the user finds desirable.

To satisfy these demands, the behavioural models used to generate the behaviour in an animated object are becoming increasingly complex. This creates a conflict when a user

20 relatively unskilled in programming wishes to modify the behaviour of an object, as they lack the expertise required to interface with the complex programs representing the behavioural model and modify the code underlying the behaviour.

Techniques to generate autonomous behaviour in an object and provide objects with

25 personality which already known in the art are limited in their usefulness. Generally, a poor compromise is reached when providing an unskilled individual with suitable means to modify an object's behaviour and supporting complex behaviour by the object. This is especially so if a user wishes to modify the behaviour of an object in real-time.

30 The present invention seeks to provide a behavioural translator which translates behaviour autonomously generated for an object from one conforming with one set of criteria to another set of criteria which obviates and/or mitigates the disadvantages

3

known in the art described hereinabove. The different sets of criteria may comprise criteria representative of different cultures and/other demographic groups.

Aspects of the invention are provided by the accompanying independent claims, in which
5   preferred features appear as dependent claims, and as set out below.

A first aspect of the invention seeks to provide a behavioural translator arranged for use by a behavioural controller of a virtual object, the translator arranged to map information representing behaviour conforming to a first demographic group to behaviour conforming
10   to a second demographic group, the behavioural translator comprising: means to receive input; means to process said input to associate the received input with behaviour by the object conforming to the first demographic group; and means to map the behaviour derived from the received input to output generating behaviour conforming to the second demographic group.
15
Preferably, the translated behaviour is generated using a method of generating behaviour for an object under the control of a behavioural controller, the method comprising the steps of: receiving input associated with one or more behavioural actions; inferring a plurality of behavioural parameter values from said input in accordance with a
20   behavioural framework arranged to generate behaviour by the object; deriving output from the inferred plurality of behavioural parameter values; and generating equivalent behaviour by the object using the output derived from the parameter values.

25   Preferably, the behavioural framework has an internally flexible structure.

Preferably, the framework comprises a hierarchy of behavioural nodes.

Preferably, the framework is dynamically flexible.
30
Preferably, input received is associated with a plurality of behavioural actions, and each parameter values inferred for is determined by a combination of said plurality of

4

behavioural action inputs.

Preferably, the input comprises a set of at least one behavioural parameter values directly associated with output which generates the behavioural action, wherein in the
5   step of inferring, at least one or more other behavioural parameter values are inferred from which further output is derived to generate equivalent behaviour to the behavioural action.

Preferably, the framework comprises a plurality of nodes, each node associated with a
10   function operating on one or more parameter values to provide output which modifies a characteristic of the behaviour of the object.

Preferably, the output produced by a function operating on one or more behavioural parameter values provides input to an animation system to generate the behaviour.
15
Preferably, the function operates on at least one behavioural parameter value assigned uniquely to the node.

Preferably, the function operates on at least one behavioural parameter value which is a
20   global parameter value available for use by any node of the framework.

Preferably, said global parameter value is associated with a mood state of the object, wherein the characteristic of the behaviour of the object provided by the output of a node of the framework is modified to indicate the mood the object is in.
25
Preferably, the node which generates output from input using a function operating on an internal parameter value associated with a personality trait affecting the characteristic of the behaviour of the object.

30   Preferably, the input is received at an output node of the framework, and the received input comprises a set of one or more parameter values directly associated with output generating the behavioural action, and in the step of inferring further parameter values, a

5

reverse mapping is performed from the set of already determined parameter values to infer a further plurality of behavioural parameter values for the internal nodes of the framework.

5   Preferably, in the step of receiving input, input is received at a global parameter node of the framework and the nodes of the network map the received input to one or more other nodes to infer a plurality of behavioural parameter values for the one or more other nodes of the framework.

10   Preferably, in the step of generating equivalent behaviour in the object, the equivalent behaviour is generated in an articulate object.

Preferably, in the step of generating equivalent behaviour in the object, the equivalent behaviour comprises facially expressive behaviour.

15

Preferably, the equivalent behaviour by the object comprises a plurality of behavioural actions performed in a predetermined sequence.

Preferably, the equivalent behaviour by the object comprises a plurality of behavioural
20   actions are performed in a random order.

Preferably, the behavioural actions are performed over a period of time.

Preferably, one or more of said plurality of behavioural actions are performed
25   simultaneously.

Preferably, the behaviour includes a behavioural action taken from the group including: eye gaze, limb movement, speech, stance, posture, tone of voice, nodding, the distance between objects, behavioural etiquette.

30

Preferably, the received input is derived from a behavioural action by the object which has been induced by direct manipulation of the object by a human user.

Alternatively, the received input is derived from a behavioural action by another object interacting with the object.

5   Preferably, the object is a virtual object operating in a virtual environment, wherein the received input is derived from a behavioural action by another virtual object interacting with the object in the virtual environment.

Preferably, the received input includes input associated with a behavioural action

10   performed by a user of the behavioural controller.

Preferably, the method further comprises the step of: translating a behavioural action received as input into a culturally equivalent behavioural action, and generating equivalent behaviour to the culturally equivalent

15   behavioural action.

Preferably, the translated behaviour is generated using a method of inferring a plurality of internal parameter values for a behavioural controller for an object, the method comprising the steps of: receiving input representing a behavioural action; inferring from

20   said received input a set of at least one output values which corresponds to an equivalent behavioural action by the object; and inferring a value for each said plurality of internal parameters from said set of at least one output values, wherein the value inferred for each said plurality of internal parameters produces output by the behavioural controller resulting in equivalent behaviour to the equivalent behavioural action.

25

Preferably, the translated behaviour is generated using a method of generating behaviour in an object, the method comprising inferring a plurality of parameter values for a behavioural controller for an object according to the above aspect, and further comprising: generating said set of output values associated with said equivalent

30   behaviour using said inferred plurality of parameter values; and causing said articulate object to perform said behaviour.

Preferably, the translated behaviour is generated using a method of controlling the behaviour of an articulate object, the method comprising the steps of: assigning a value to a behavioural parameter set associated with a behavioural characteristic of the object using a behavioural design interface arranged to provide input to a behavioural controller

5   for the object, each said behavioural parameter set comprising at least one parameter affecting the behavioural characteristic; associating each parameter in the parameter set with a parameter value obtained by performing a function on the assigned value with a default value defined by a behavioural profile; inputting the parameter value to the behavioural controller for the object; inferring from said input, output generated by the

10   behavioural controller; associating the output with a behavioural action by the object; and causing the object to perform the behavioural action.

Preferably, the function is an identity function.

15   Preferably, the translated behaviour is generated from behaviour which has been generated using a method of directly manipulating an object to control its behaviour, the method comprising the steps of: manipulating the object to perform a behavioural action; providing input representing the behavioural action to an output node of a behavioural framework, the output node being also arranged to provide output which is used to

20   generate equivalent behaviour by the object, mapping the input received by the output node of the behavioural framework within the framework to derive a set of at least one parameter values for other behavioural nodes of the framework; inferring from the set of at least one parameter values derived a set of output values which will generate other equivalent behaviour by the object.

25

Preferably, the behaviour of the object is generated in real-time in response to receiving input associated with a behavioural action.

Preferably, the object is a virtual object provided in a virtual environment.

30

Alternatively, the object is a robotic object. The object may be a toy.

8

Preferably, the translated behaviour is generated using apparatus comprising a
behavioural controller arranged to generate behaviour in an object, the controller
comprising: means to receive input associated with a behavioural action; means to infer
a plurality of behavioural parameter values from said input in accordance with a

5   behavioural framework arranged to generate behaviour by the object; means to derive
output from the inferred plurality of behavioural parameter values; andmeans to
generate equivalent behaviour by the object using the output derived from the parameter
values.

10   Preferably, the means to generate equivalent behaviour comprise means to forward the
output derived from the parameter values to an animation system arranged to operate on
the output to cause the appropriate behaviour to be animated by the object.

Preferably, receiving means include means to receive as input at least one parameter

15   value from a source external to the behavioural framework of the object.

Preferably, the means to infer a plurality of behavioural parameter values comprises a
framework of nodes, each behavioural node arranged to map at least one input
parameter value to at least one output parameter value.

20

Preferably, at least one node is arranged to map at least one parameter value taken from
the group including: a parameter defined for each node within the behavioural
framework; a parameter defined within each node of the behavioural framework; and, a
parameter defined externally to the behavioural framework.

25

Preferably, said means to receive input is arranged to receive input from a behavioural
design interface, the behavioural design interface comprising:means arranged to allow
the assignment of a value to a behavioural parameter set comprising at least one
behavioural parameter defined according to the behavioural framework of the object; and

30   means arranged to operate on the value assigned to the behavioural parameter set by a
predetermined function to determine the value of the internal parameter.

9

Preferably, the object is a virtual object arranged to operate within a virtual environment.

Preferably, output from the behavioural controller is provided in a form suitable for being received as input by a behavioural controller of another object.

Preferably, the behavioural controller further comprises a translation element for mapping received input derived from behaviour consistent with a first culture to input consistent with a second culture.

Preferably, the behavioural controller further comprises a translation element for mapping behavioural output consistent with a first predefined culture to behavioural output consistent with a second predefined culture.

Preferably, the object is a virtual object arranged to operate within a virtual environment is taken from any one of the group of virtual environments consisting of:a virtual computer game, a virtual on-line meeting, an on-line game, an on-line chat-room, an avatar hosted meeting; an avatar counselling meeting; an avatar based mediation environment; an avatar based sales environment; an on-line collaboration environment; an on-line customer relationship management environment.

Preferably, the translated behaviour is generated using apparatuscomprising a behavioural design interface, the interface comprising:means arranged to allow the assignment of a value to a behavioural parameter set, the parameter set comprising at least one parameter value associated with a behavioural characteristic of the object, wherein the value assigned using the interface is provided as input to the apparatus as claimed in any one of the above aspects.

Preferably, the input means is arranged to receive input comprising a set of at least one behavioural parameter values directly associated with output which generates the behavioural action, wherein the means to infer is arranged to infer at least one or more other behavioural parameter values from which further output is derived to generate equivalent behaviour to the behavioural action.

Preferably, the input means receives input comprising at set of at least one behavioural parameter values directly associated with output corresponding to a direct manipulation of the object.

5

Preferably, a user provides the input to the apparatus. Alternatively, a software agent provides the input to the apparatus.

Preferably, the translated behaviour is generated using a device arranged to have a suite

10 of at least one computer programs stored thereon, the suite of at least one computer programs being executable on the device so as to cause the device to function as the apparatus defined in any of the above aspects.

Preferably, the translated behaviour is generated using a network comprising a plurality

15 of said devices arranged to be capable of communicating with each other, the each device being arranged to remotely access at least part of the suite of at least one computer programs, to enable objects operating within the environments of said other devices to be controlled by the suite of at least one computer programs.

20 Preferably, the translated behaviour is generated using a method of generating behaviour in an object under the control of a behavioural controller comprising a framework of nodes, the method comprising the steps of: at least one node receiving input associated with a behavioural action; each said at least one node mapping received input to output; inferring a plurality of behavioural parameter values for other

25 nodes in the framework using said output; mapping the received input using said inferred behavioural parameter values to provide output by the behavioural controller which generates equivalent behaviour by the object.

Preferably, the translated behaviour is generated using a method of generating

30 behaviour in an object under the control of a behavioural controller, the method comprising the steps of: receiving input associated with a behaviour action; mapping said received input to a set at least one output values which corresponds to equivalent

11

behaviour by the object; inferring a plurality of behavioural parameter values from said set of at least one output values in accordance with a behavioural framework arranged to generate behaviour by the object; and generating equivalent behaviour in the object using said parameter values by loading these into the behavioural controller.

5

Preferably, the parameters inferred are time-varying.

Another aspect of the invention relates to a computer program product comprising a suite of one or more computer programs arranged to execute a set of instructions to perform

10   steps in the above method aspects of the invention.

Advantageously, more complex behaviour may be translated even where the complex behaviour is provided by relatively unskilled users as the equivalent behaviour by the object may comprise a plurality of behavioural actions performed in a predetermined

15   sequence.   Even more advantageously, time-varying behavioural parameters may be incorporated into the behavioural controller to provide more realistic autonomously animated behaviour.

Advantageously, by having the ability to infer from input internal parameter values which

20   can then be used to generate equivalent actions by the object, the behavioural translator according to the invention is able to use the method of generating behaviour to utilise relatively simple data input provided by a user or by another application to generate complex behaviour in the object.

25   Advantageously, complex behaviour may be generated which may comprise more than one behavioural action, two or more of which may be performed simultaneously. For example, the behaviour of an articulate object such as an avatar may be controlled so that both posture and eye gaze behaviours of the avatar are performed simultaneously. For example, the avatar could automatically look at an object when picking the object up,

30   or alternatively, as another example, if an avatar is timid in response to another avatar being aggressive, the timid avatar may change both posture and eye gaze to hunch his/her shoulders and keep his/her eyes downcast.

12

Advantageously, the invention provides a means to enhance a chat room experience for a user by increasing the amount of expression the users can convey in the virtual medium of the chat room, and providing a method of translating expressions between sets of criteria conforming to different demographic groups The presence of body language, in particular when conveyed by the simultaneous performance of two or more behavioural actions such as posture and eye gaze, renders avatars more life-like and human and improves the quality of their communication. However, as more complex behaviour is generated, subtle demographic group and/or cultural differences in behaviour which enhance the automated behaviour generated may not have the same context in other demographic groups/cultures.   Advantageously, the method of generating behaviour described herein enables behavioural actions to be generated according to the cultural and/or demographic group of an observer of the object, and where an object is seen by observers from different demographic groups/cultures, the behaviour is automatically translated to equivalent behaviour conforming with criteria for the differing cultural/demographic groups. This may result in different/less or additional behavioural actions being performed by an object when being perceived by an observer belonging to one demographic/cultural group from the behavioural actions perceived by an observer in another group.

Thus, the invention advantageously enables a group of avatars to co-ordinate their behaviour so that each avatar reacts to the body language of other avatars in an appropriate manner.

The preferred features as set out in the dependent claims may be suitably combined with any of the above aspects in any appropriate manner apparent to those skilled in the art.

Preferred embodiments of the invention will now be described, by way of example only, with reference to the accompanying drawings in which:

Figure 1A shows schematically a first embodiment of the invention;

13

Figure 1B shows schematically an enlarged view of the behavioural design user interface shown in Figure 1A;

Figure 2A shows in more detail the neutral posture of both avatars as shown in Figure

5   1A;

Figure 2B shows the posture of the two avatars when the Bill avatar has a high machismo;

10   Figure 2C shows the posture of the two avatars when the Bill avatar has a high flirtatiousness;

Figure 2D shows the posture of the two avatars when the Bill avatar has a lower flirtatiousness;

15

Figure 2E shows the posture of the two avatars when the BOB avatar has high levels of being putUpon and listening, and a low level of domSub;

Figure 3 shows a schematic overview of the behavioural architecture according to the

20   invention;

Figure 4 shows the hierarchical layers of a behavioural model according to an embodiment of the invention;

25   Figure 5 is a schematic diagram of a simplistic framework for the behavioural controller for an object according to an embodiment of the invention;

Figure 6 is a schematic diagram of a behavioural node in the framework of an architecture such as that shown in Figures 5 and 7;

30

Figure 7 is a schematic diagram of a more sophisticated framework for a behavioural controller than that shown in Figure 5 for an object according to another embodiment of

14

the invention;

Figure 8 is a flow diagram schematically indicating real-time steps in a method of generating behaviour in an object according to the invention, and off-line profile and

5   adjective design;

Figure 9A is a flow diagram indicating how the behavioural framework generates equivalent behaviour according to the invention;

10   Figure 9B is a flow diagram indicating how a user can manipulate an avatar's body movement to reassign the values assigned to specific behavioural parameters of the behavioural controller;

Figure 10A shows schematically an avatar having performed a greeting behavioural

15   action;

Figure 10B shows schematically an avatar having performed an equivalent greeting behavioural action to that shown in Figure 10A; and

20   Figure 10C shows schematically an avatar having performed yet another a greeting behavioural action equivalent to those shown in Figures 10A and 10B.

The best mode of the invention as currently contemplated by the inventors will now be described with reference to the accompanying drawings. It will be apparent to those of

25   ordinary skill in the art, however, that the description of the invention is by way of example only, and that the invention is not intended to be limited only to the embodiments described.

The behavioural translator according to one embodiment of the invention, is implemented

30   using the methods and apparatus for generating behaviour which are described herein below with reference to the accompanying drawings.

15

Referring now to Figure 1A of the accompanying drawings, a first embodiment of the invention comprising a tool for generating, inferring and designing body language for avatars and virtual characters is shown. This embodiment of the invention is of particular use for controlling the behavioural characteristics of virtual objects in internet chat-rooms

5   and computer game-type environments and the like.

In Figure 1A, a computer device 1 is shown. The computer device 1 is assumed to be operated in use by a human user (not shown) who may not have a high level of computer programming skill. The term computer device is used to imply any device having data

10   processing ability which can be attached to a visual display, for example, games consoles, personal digital assistants, as well as mobile-communications devices such as mobile telephones etc.

The computer device 1 is provided with visual display means 2, for example, a monitor,

15   having display 3. Any suitable navigation means may be employed by the user to navigate the display 3, for example a mouse or keyboard (not shown). Other embodiments may include navigation tools such as styluses, track-pads, and joysticks which may be used in an equivalent manner.

20   Display 3 includes a window 4 within which a virtual environment application is running. A virtual environment is displayed in window 4 which contains virtual objects. For clarity, only two virtual objects will be discussed in the context of this preferred embodiment of the invention. As shown in Figure 1A, the two virtual objects are articulated objects comprising two avatars 5,6 capable of being animated. Avatar 5 is also referred to

25   herein as Bob and avatar 6 is also referred to as Bill. Whilst Bill and Bob have an articulated form in the embodiment shown in Figure 1A, it is possible for the virtual objects to be non-articulated, for example, to comprise faces which contort to display emotions etc. References to behaviour therefore include facially expressed behaviour and any form of behaviour by an object, regardless of the form of the object.

30

Whilst virtual objects 5,6 are arranged to be capable of being animated semi-autonomously (in which case they will require at least some input from the user), it is

16

possible for at least one of the virtual objects 5,6 to be programmed off-line to function autonomously using a behavioural controller according to other embodiments of the invention. In the context of the invention, off-line refers to programming the behavioural controller when the virtual objects are not performing animated behavioural in their

5   virtual environment, or when such behavioural is paused or interrupted whilst off-line programming occurs before being resumed.

Returning now to Figure 1A, the virtual objects 5,6 in the virtual environment comprise two human avatars. However, it will be appreciated by those skilled in the art that the

10   form a virtual object may take can vary considerably depending on context. For example, depending on its role a virtual object may be an animal or any other articulate object capable of being animated. The term articulate is defined here as being composed of parts which are capable of being moved relative to each other, for example, limbs and/or joints.

15

Display 3 also shows a behavioural design user interface 7. The behavioural design user interface 7 comprises a separate window to the virtual environment window 4. The behavioural design user interface 7 enables a user to provide input to a behavioural

20   controller to generate behaviour by one or more of the virtual objects 5,6. The behavioural design user interface application may comprise part of an application including the behavioural controller or it may be provided as part of a separate software application arranged to interface with an application including the behavioural controller. The behavioural controller comprises a set of connected nodes arranged according to a

25   predetermined behavioural framework, each node in the framework mapping inputs to outputs based on a number of parameters. The behavioural controller is described in more detail herein below.

In Figure 1A, the user has control over both virtual objects 5,6 by appropriately selecting

30   behavioural parameter values using sliders 8 in tracker bars 9 of the behavioural design user interface. Only the tracker bar and slider for the machismo behavioural parameter set has been numbered in Figure 1A for clarity. However, referring now to Figure 1B, an

17

enlarged view of the behavioural design user interface of Figure 1A is shown.

THE BEHAVIOURAL DESIGN INTERFACE

5   Figure 1B shows the behavioural design user interface 7 of Figure 1A in more detail. As
     Figure 1B shows, the behavioural design user interface 7 provides a user with a range of
     menu choices for each of the virtual objects Bob and Bill which are interacting in the
     virtual environment window 4 of Figure 1A. In alternative embodiments of the invention,
     the user may have less direct control over the behaviour of other virtual objects in the
10   environment, but for simplicity, it will be assumed here that the user wishes to have direct
     influence over both Bob and Bill. Even where no direct control over another virtual object
     is provided, however, the user can still influence the behaviour of other virtual objects
     indirectly as will be explained in more detail.

15   For clarity in Figure 1B, only slide bars 8,10 for selecting the value of the behavioural
     parameter set labelled flirtatiousness in tracker lanes 9a, 11a have been numbered.

     The behavioural design user interface 7 as shown in Figure 1B comprises two
     behavioural profiles 9a to 9h and 11a to 11h. Each behavioural profile shown comprises
20   a set of "adjectives" or equivalently behavioural parameter sets which are associated with
     a particular behavioural characteristic of the virtual object. As shown in Figure 1B, the
     behavioural parameter sets include flirtatiousness 9a, 11a, friendliness 9b, 11b,
     machismo (9c, 11c), otherLiking (9d, 11d), sad (9e,11e), selfImportance (9f, 11f),
     shyness(9g, 11g) and superioritytoOther (9h, 11h).

25
     A user can construct other behavioural profiles containing a different selection of
     behavioural parameter sets. It is also possible to modify the properties of each of the
     plurality of individual behavioural parameters which comprise a behavioural parameter
     set. The design of the behavioural parameter set is defined by the behavioural
30   framework of the behavioural controller to collectively modify a predetermined
     behavioural characteristic. Accordingly, parameter set design is a task which requires
     more skill than the simple selection of what behavioural parameter sets will form a

18

behavioural profile. The parameter set can comprise just a single parameter. For example, a global parameter whose value can affect the outputs of all behavioural nodes within the behavioural framework or an internal parameter whose value affects the output of only a single behavioural node of the framework.

5

Returning to Figure 1B, the track lengths of the tracker lanes 9a to 9h and 11a to 11h represent a range of possible values a user can select by positioning the slide bars 8 or 9. It will be appreciated that only slide bars for the first flirtatiousness behavioural parameter set have been labelled in Figure 1B for clarity. In other alternative

10   embodiments, other value selection means may be used, for example, radio buttons, drop-down windows etc, or directly inputting control data. However, the use of a sliding bar and tracker is particularly preferred as this enables a continuous range of values to be easily assigned to a parameter without unduly distracting the user from the main screen display virtual environment window 4.

15

The values shown in Figure 1B, provide the virtual objects 5,6 with a neutral stance corresponding to each displayed behavioural parameter set been assigned the value zero by the user. This is shown more clearly in Figure 2A, which shows the neutral posture of the two virtual objects in shown in Figure 1A and the corresponding

20   behavioural parameter set values shown in Figures 1A and 1B.

Figure 2B shows how a user has directly influenced Bill's behaviour by moving the slider in the machismo tracker bar for virtual object 6 (Bill), and indirectly influenced Bob's behaviour. In Figure 2B, the user has assigned a value of 14 using the slider in the

25   machismo behavioural parameter set tracker lane, and Bill displays behaviour generated by the behavioural controller for Bill. The behavioural controller of Bill has inferred what equivalent machismo behaviour is required from the user input by taking this input and propagating it through the behavioural framework to assign appropriate values to the behavioural output nodes of the framework.

30

The behavioural controller for Bill has inferred from the assigned machismo value of 14, equivalent behavioural action which in the embodiment shown in Figure 2B comprises

19

Bill maintaining a machismo posture. A variety of machismo postures may result, and also other machismo behaviour such as machismo gestures, eye gaze, stance, facial expressions etc. The behavioural controller of Bill also outputs one or more values to the behavioural controller of Bob which reflect Bill's machismo behavioural action(s). This

5   output comprises a behavioural action to be performed by Bob indicating his response to Bill's machismo behaviour.

The behavioural controller of Bob receives the input derived from Bill's behavioural action which enables Bob's behavioural controller to effectively interpret the body language

10   conveyed by Bill's behaviour. This input can represent a behavioural action comprising a response to Bill's machismo behaviour, for example, one or more parameter values which will modify the behaviour generated by Bob's behavioural framework. Alternatively, Bob may just receive an indication of parameter values indicative of Bill's behaviour, and Bob's behavioural controller could infer from the parameter values which

15   are provided by Bill, what output values would correspond to this response. Bob's behavioural controller is then able to infer from these output values other behavioural parameter values which in Bob's behavioural framework would generate equivalent behaviour to the response, generates output using these behavioural parameter values, and then performs this behaviour. As shown in Figure 2A, this comprises Bob adopting a

20   submissive posture in front of Bill.

In the manner described herein above, a user is indirectly able to modify the behaviour of virtual object 5 by changing a behavioural parameter set value for the virtual object 6. Bob has assumed a more submissive stance without the need for the user to directly

25   move Bob's tracker bar to a submissive position. The behavioural controller for one virtual object (Bob) has interpreted the body language conveyed by the behaviour of the other virtual object (Bill). This has been achieved by the behavioural controller of the first object generating output received by the behavioural controller of the other object which conveys information indicating the body language. Suitable internal behavioural

30   parameter values are then inferred by the behavioural controller for the other virtual object , which modifies the behaviour of the other virtual object. It will be apparent to those skilled in the art, that it is possible for the body language information to be

20

conveyed to more than one virtual object, provided each virtual object has an appropriate behavioural controller arranged to received input .

5   In the embodiment of the invention shown in Figure 2A, the parameter values of the behavioural framework corresponding to the character Bob (virtual object 6) are not directly affected by the values assigned to Bill, and as the displayed slider values for Bob do not change their values. However, in other embodiments of the invention it is possible for the slider values to be automatically updated to reflect their new values.

10   Figure 2C shows how in one embodiment of the invention, the body language of one of the virtual objects can be conveyed to the other virtual object to set up an appropriate behavioural response. In Figure 2C, following on from the behaviour shown in Figure 2B, Bill's user has assigned a value of 1 to machismo and has assigned a value of 10 for flirtatiousness. The behavioural controller for Bill now generates automated behaviour,
15   such as animating Bill to reach out and touch the arm of Bob as shown in Figure 2C. Bob's behavioural controller interprets Bill's body language as indicating he is less machismo and more friendly and that the action of reaching out to touch Bob is therefore friendly and not hostile. This can be achieved for example, by using the input Bob's behavioural controller has received to interpret the parameter values provided by Bill's
20   controller as indicating Bill is being friendly. Consequently, instead of displaying defensive behaviour (for example) which might ensue if Bill was touch Bob in a hostile way, Bob instead adopts a less defensive posture, and leans slightly forwards, rather than maintaining a submissive posture. Seeing that Bob is less defensive, the user can now set the level of flirtatiousness and machismo for Bill to lower values, as is shown in
25   Figure 2D, which produces more friendly behaviour by both avatars.

As shown in Figures 2A to 2D, at no point does a user have to indicate specific body movements or key in text to indicate a behavioural action, the behavioural controller generates appropriate behaviour autonomously in response to the assigned behavioural
30   parameter set values in real time.

Advantageously, by providing the behavioural design user interface 7 as a window-type

software application, the extent of the interface on view can be varied according to a users requirements. In Figure 2E, the behavioural profile for Bob is visible in the behavioural design user interface window 7 and a different behavioural profile is shown for Bob which includes different behavioural parameter sets from those shown in the

5   behavioural profiles in Figures 2A to 2D.

In Figure 2E, Bob has been assigned by the user a high value for the listening behavioural parameter set, an equally high value for the putUpon behavioural parameter set. Other behavioural parameter sets shown in Fig. 2E include touchyFeely, domSub,

10   etc. Bob has been assigned a relatively low value of domSub. These values have been used by the behavioural controller of Bob to produce behaviour which is appropriate to these values, accordingly, Bill has an erect posture and Bob has a more submissive posture.

15   Figure 3 of the accompanying drawings shows schematically how the behavioural controller of the invention generates the behaviour of an object according to a behavioural model 30. The behavioural model 30 defines a specific behavioural framework 31 for the object which governs how inputs received by the framework are used to generate certain behaviours such as, for example, posture 32, gesture 33, facial

20   movement 34, eye gaze 35. The behavioural framework 31 comprises a number of nodes and is described in more detail later (see Figures 5 and 7 for examples of behavioural frameworks).

The inputs to the framework 31 can be derived from a variety of external and internal

25   sources. For example, from external contexts/events 36, from other characters 37, from other mood cues 38, from a predefined profile 39. The framework 31 itself can also be used to design certain profiles 40, the resulting profiles 39 then providing input to the framework 31 to modify the values assigned to one or more behavioural parameters of the framework nodes.

30

Where a user directly manipulates an avatar or directly inputs values using the behavioural design user interface 41, input can be provided directly to the framework 31

22

by the values assigned by the movements to the posture 32, gesture 33, facial 34, and eye gaze 34 behavioural output nodes of the framework. The framework then infers appropriate values for behavioural actions and the user interface/tracking system 40 then produces appropriate actions using the animation system.

Collectively, the values output by the posture, gesture, facial, and eye behavioural nodes are used to produce appropriately animated behaviour using an animation subsystem 41. The animation subsystem used can be provided by any appropriate animation application, for example a computer game engine such as the Quake engine or a scene graph based computer graphics system such as SGI's Open Inventor library.

THE BEHAVIOURAL MODEL ARCHITECTURE

Referring now to Figure 4 of the accompanying drawings, the hierarchical structure of a layered hierarchical behavioural model according to the invention is shown schematically.

Figure 4 shows a preferred embodiment of the invention in which the functionality of the behavioural model comprises five layers: a first layer which functionally relates to the design of behavioural nodes in the framework defining the behavioural model 51; a second layer which functionally relates to the design of the behavioural architecture itself 52a and to "content creation" 52b (which relates to the creation of actual animations etc. corresponding to the output behaviour, by skilled artists); a third layer which relates functionally to the design of behavioural adjectives (equivalently, behavioural parameter sets) 53; a fourth layer which functionally relates to parameter value selection mechanism, for example, as shown in Figure 4 the slider function 54a, to the design of behavioural profiles 54b, and to the direct provision of input into the model by direct manipulation of a virtual object 54c; and finally a fifth layer which relates to real time control 55.

In other embodiments of the invention, more layers of complexity can be provided in the behaviour design and control architecture, however, five layers is the minimum required by the preferred embodiment of the present invention if real time control is to be

supported.

The level of specialised knowledge and/or the amount of information required to interface
with a layer of the behavioural model generally depends on the specific feature of the
framework or function a user is seeking to modify. For example, the interface to the
upper layers of the model (e.g. layers 4 and 5) require relatively little specialised
knowledge on the part of a user, i.e., anyone can perform real time control of a virtual
object according to this aspect of the invention. However, a user wishing to design a
node type (i.e., interface with level 1 of the behavioural hierarchy) is likely to be a
specialist programmer.

The behavioural model shown in Figure 4 differs from the known multi-layered behaviour
design and control architecture of Scerri & Ydrèn (see below), in both the number of
layers (which increases the complexity) and the inclusion of real time control in the model
(for example, see Scerri and Ydrèn [End User Specification of RoboCup Teams,
RoboCup-99, Robot Soccer World Cup III, Springer-Verlag Lecture Notes in Computer
Science(2000)] for more details of this simple multi-layer architecture). Other
distinguishing features, in addition to having a more complex hierarchical structure, and
the ability to implement real-time control functionality, include the provision of a
behavioural parameter inference scheme which enables behavioural parameter values of
the behavioural framework to be internally inferred from inputted parameter values.
Thus, when a user inputs a set of one or more behavioural parameter values associated
with a behavioural characteristic, or manipulates the object to produce a specific
behavioural action, the received input can be used to generate other equivalent
behaviour comprising one or more behavioural actions. This behavioural parameter
inference system is described in more detail later herein below.

In Figure 4, the design node type layer 51 relates to the design of output nodes that
interface with an animation control system. Typically, the output of an output node is
used by other sections of the node to animate the virtual object. The animation system
contains a number of parameters that control the behaviour that it produces. An output
node has one output for each parameter of the animation system and that parameter is

24

directly set to the value of the output. For example, in the case of a posture node a new posture is generated as a combination of a set of basis postures, based on a weighting for each basis posture. The postures are blended together with a motion combination system in proportion to their weights. The posture node has an output for each posture that corresponds to its weight. Other output behaviours would have more complex mappings between parameters and behaviour. For example, an output node can be created by a programmer (generally quite skilled in their art) creating a sub-type of the node type and then, by adding the new type to the framework of the architecture at run time, the node can be used for reading in a behavioural controller definition from a file. In other embodiments of the invention, the output node adapt their output to suit the animation system being used.

A user would generally need to be trained to be familiar with the behavioural framework before modifying the architecture design 52a or content creation 52b features of layer 2 of the behavioural design model. Layer 2 comprises the framework for creating virtual objects for a particular application. It includes the design of the behavioural controller and the design of content for an output node. For example, an output node can be designed to produce behaviour that is based on pre-existing motion or other content. Many output behaviours will be based on some pre-existing animations and similar content, for example, a posture model is based on a set of pre-existing postures and a facial expression module would be based on a set of pre-existing facial expressions. These can be created by a skilled designer using commercial 3D modelling tools.

The design of the behavioural controller is typically specified by a design specification file, for example an XML file, or other suitable file-type (possibly a specially designed file-type), which can be edited by hand. As the behavioural controller has a graph structure, a simple graphical editing tool may be provided for editing the design specification file in alternative embodiments. Once the design specification file has been edited it can be . complied into a controller using the behavioural framework described above.

The adjective design layer 53 and sliders 54a, profile design 52b, direct manipulation 52c, and real time control features of layers 3,4 and 5 in Figure 4 are arranged to enable

25

a generally unskilled user to customise the behaviour of an object. In particular, a user is able to interface with layers 3 and 4 by means of designing a behavioural profile, as described in more detail herein below with reference to Figure 8 of the accompanying drawings.

THE BEHAVIOURAL CONTROLLER

Referring now to Figure 5 of the accompanying drawings, a framework 60 of a behavioural controller for a virtual object according to a first embodiment of the invention is shown. In Figure 5, the framework 60 of the behavioural controller comprises a number of computational nodes which map input from one or more sources to one or more outputs. The nodes within the framework 60 include nodes providing external input 61, for example, input which may be derived from the behaviour of other virtual objects; global parameter nodes 62 which provide global framework parameters and their associated input values which is accessible by all nodes in the framework (either directly or indirectly); behavioural nodes 63a,63b, 63c, 63d, which are identified by a name and which are associated with one of more values internal to the specific node; and output nodes 64,65, which may comprise external output nodes 64 which output parameter values which can be used externally (e.g. for use by other virtual objects' behavioural controllers), or behavioural output nodes 65 which provide parameter values which are used by the behavioural animation mechanism to produce the actual desired animation of the virtual object providing the appropriate behaviour. From a programming perspective, each parameter consists of a name-value pair, e.g., a textual name with an assigned numeric value. The precise architecture of the behavioural model used will determine the form of the framework 60 of the behavioural controller.

In Figure 5, the framework 60 comprises a number of behavioural nodes 63a,b,c,d whose function is to map a number of inputs to a number of outputs based on a number of parameters. Figure 5 shows schematically how external inputs 61 and global parameter inputs 62 collectively provide input to behavioural nodes 63a,b,c,d. Nodes 63a,d additionally receive input from nodes 63b,c.

External input 61 comprises high level information about the environment and other objects, for example, the degree to which an other character is being friendly, or submissive.

Global parameter input 62 comprises high level attributes of the virtual object that influence its behaviour and which modify the specific behaviour determined by each behavioural node. For example, the global parameter values may comprise a characteristic such as the mood or attitude of an object, e.g., happy or friendly. Referring briefly back to Fig. 1B, several behavioural parameter sets are labelled to indicate various global parameters, such as how friendly a character is or how shy.

Each global parameter name-value pair inputted to a behavioural node 63a,b,c,d within the behavioural controller framework generates one or more numerical outputs. These numerical outputs are then passed on as either external output by external output nodes 64 or are associated with behavioural output by output nodes 65.

External output 64 comprises information equivalent to the external input, for example how friendly or submissive the virtual object is being Parameter name-value pairs provided as external output convey body language information. When this external output is received by other virtual object(s), it enables internal behavioural parameters of the other virtual object(s) to be inferred which modifies the behaviour of the other virtual object(s). The external output by one controller is correlated with the external input provided to the behavioural controller(s) of other virtual object(s) by matching name-value pairs having the same name.

Each behavioural output node 65 produces output corresponding to a behavioural action. From a programming perspective, a behavioural output node 65 comprises a sub-type (in an object-oriented sense) of a behavioural node 63a,b,c,d and performs a similar map of input to output to map from parameters to behaviour. A behavioural output node 65 produces output that can be used to animate the character by other parts of the output node. For example, in a posture output node, there are a set of basis posture from which new postures are generated, and a parameter for each basis posture. Actual

representation of a posture is stored in terms of an object's joint angles (as Euler angles). A new posture is generated by performing a weighted sum on the angles corresponding to the basis posture using the parameters of the postures as weights. These generated angles are passed directly into the corresponding transforms in the underlying geometric representation.

THE STRUCTURE AND FUNCTION OF BEHAVIOURAL NODES IN THE FRAMEWORK

Referring now to Figure 6, an enlarged view of behaviour node 63d of Figure 5 is shown. Figure 6 shows schematically how input from a plurality of differing sources may be used by a behaviour node. As shown in Figure 6, behavioural node 63d is capable of receiving up to three different types of input which are mapped in a forwards direction by the behaviour node to one or more outputs based on its internal parameter set. In Figure 6 input to behavioural node 63d can come from the output from another behavioural node, e.g. nodes 63b,c; from the input 62 provided by one or more global parameter name value pairs; and/or from external input 61 from a source outside the framework. The external input 61 may be generated by another object with which the object is interacting, according to a predetermined set of interaction rules. The external input to the behaviour node may be modified by the node. For example, input may be ignored, or limited to a maximum or minimum value if the input extends beyond an acceptable range. Alternatively, if external input represents an action performed in accordance with a different culture to that of the user, the external input may first be appropriately modified to ensure that external input corresponding to the appropriate behaviour in the user's own culture is in fact used by the framework to modify the response by the user's virtual object.

Figure 6 also shows how a behavioural node can reverse its functionality and perform a reverse map. A reverse map is performed whenever input is received by the framework at a level which corresponds to the output of the behavioural nodes. This can occur, for example, when a user directly manipulates an object as this provides input to the framework at a level equivalent to the output to the behavioural nodes 65. This received

28

"output" is then the starting point for a reverse map through the framework, each internal behavioural node having its parameter values inferred in a manner described in more detail later herein below, until eventually even the global parameter values for the framework which would produce the received "output" are determined.

In either a forwards or backwards direction, each behavioural node in the framework is capable to map one or more inputs to one or more outputs based on a number of parameters, according to the function of the node.

FORWARDS MAP

In the case of the forwards map, the outputs provided by the behavioural controller for an object given as the sum of a number of terms $O_i = \sum_j T_{ij}$, where each term Tij is the product of a number of factors $T_{ij} = \prod_k F_{ijk}$, where each factor is either an internal parameter or an input of the node. As indicated above, the inputs to a node may originate as outputs from another node, be parameters assigned globally to the entire architecture or be external inputs, coming from another architecture (i.e., from another object's behavioural controller).

For example, a character could be set with a global parameter "friendliness" with a value 1.2 (indicating that the character is naturally friendly), it would also receive an external input "pleasantness" from another character with a value of 1.5 (indicating that the other character is being pleasant). These would be multiplied together in a node to produce an out put "close" with a value 1.8 (indicating that the character should adopt a close posture to the other character). This output would then be passed to other nodes which might determine that the character should achieve this by combining two basis postures, leaning forward and orienting towards the other character. The weights for these two postures would be calculated from the "close" output and passed to the animation system which would generate the new posture.

Referring now to Figure 7, a schematic view is provided of a framework for a behavioural controller according to a second embodiment of the invention is shown in more detail. In Figure 7, the behavioural nodes include immediacy equilibrium, dominance factor, immediacy difference, responsiveness, dominance difference, affiliation, status, pleasantness, proximity, space fling, and relaxation. Behavioural output nodes 65 include facial expression, high level posture nodes, eye gaze, gesture, and posture. Other embodiments may include more behavioural output nodes such as speech tone, speed, accent, etc.

Whilst the complexity of the framework shown schematically in Figure 7 is higher than the simplistic framework shown in Figure 5, more complex frameworks may be constructed to suit specific applications and embodiments of the invention. To enable a user to modify the behaviour of an object which is generated according to its behavioural framework, a behavioural adjective comprising a set of one or more behavioural parameters is constructed as this greatly simplifies the level of input the user is required to supply.

Examples of behavioural adjectives include those shown in Figure 1B where the behavioural profile for each virtual object 5, 6 includes the following adjectives: flirtatiousness, friendliness, machismo, otherLiking, sad, selfImportance, shyness, and superioritytoOther. Each of these behavioural adjectives comprises a behavioural parameter set, and is generally represented in the behavioural design user interface by a simple textual name or phrase. Typically an "adjective" name intuitively describes that aspect of behaviour the "adjective" modifies, to facilitate recognition by a user. Each parameter in a behavioural parameter set can be assigned an initial or default value, which can be operated on by a function, and may be operated on in conjunction with any externally inputted value. For example, the function may be a simple linear algebraic function, or simply to scale any value assigned by a user to a behavioural parameter set by a predetermined amount. Alternatively, the function may be an "identity" function, returning just the value inputted.

The framework shown schematically in Figure 7 represents an embodiment of the invention for performing various aspects of non-verbal, intra-personal behaviour. The behavioural controller enhances interactions between virtual objects, for example, characters in an on-line meeting or computer game/chat-room scenario and can make them appear more believable. In particular, the framework is useful for semi-autonomous avatars (i.e., where the user does not specifically dictate each behavioural action of an avatar).

As many of the uses of avatars involve intra-personal behaviour, appropriate non-verbal behaviour greatly enhances their use. In Figure 7, the behavioural controller encodes a number of intra-personal attitudes and potentially controls a number of outputs, which produce animation, such as posture and eye-gaze behaviour. The control system is based on theories proposed by Argyle (Michael Argyle (1988)*Bodily Communication* 2nd edition, Routledge) and by Mehrabian (Albert Mehrabian (1972)*Nonverbal Communication*, Aldine-Atherton). Argyle proposes two aspects of interpersonal relations that have the greatest effect on non-verbal behaviour, intimacy and dominance-submission. These can be modelled as homeostatic motivations as described below. Related to these Mehrabian proposed three dimensions of non-verbal activity affiliation (liking), displayed in such things as smiling, physical closeness and touching, potency/status, displayed by relaxation or tension and responsiveness, displayed by general physical activation. The responsiveness dimension is optional and is not implemented in the behavioural controller shown in Figure 7, which is otherwise based on these quantities.

In Figure 7, as Argyle's dimensions of Immediacy and Dominance are closely associated with Mehrabian's dimensions of Affiliation and Status they are associated in the architecture. Mehrabian's dimensions are modelled as behavioural nodes that are directly determined by the Immediacy and Dominance agents. Immediacy and dominance are modelled as homeostatic motivations.

A desired value for a variable is calculated by the Immediacy Equilibrium and Dominance Factor nodes based on the global parameters of the virtual object and external inputs

from any other virtual object. Factors that increase the desired intimacy include are how
friendly the character is, how much it likes the other character (global parameters) and
how pleasant the other character is being (external input). Factors that decrease it are
how shy the character is and how dominant the other character is being. All these factors
have weightings that can vary from character to character based on their profile. The
desired dominance factor is a desired difference in status between the two characters
which also depends on a number of other factors. The Immediacy difference would be
the difference between the desired immediacy and the actual immediacy, which is
determined by how intimate the other character is being (an external input) If the actual
immediacy is I_a and the desired immediacy is I_d, the immediacy difference is:

The equation for dominance is similar though the factors are course of different. A third
dimension of behaviour responsiveness, is implemented in other embodiments of the
invention.

In this embodiment of the invention, the behaviour of the character is defined in terms of
high-level types of behaviour: pleasantness, proximity, space filling, relaxation. These act
as intermediaries between the motivational levels of the hierarchy and the action
producing levels. Pleasantness is a general pleasant demeanour such as a smiling face
while the opposite might be frowning or aggressive gestures. It is an expression of
affiliation (like or dislike). Pleasantness does not have many expressions in posture but
an example is the "head cock" where the character tilts its head to the side when with the
other character, this is normally interpreted as a friendly posture. Proximity is social
distance (closeness), including physical distance but also such things as body orientation
or amount of mutual gaze. Low social distance is a result of high affiliation. It is
expressed in posture in a number of ways such as leaning forward or touching the other
character. High social distance is the opposite and can be expressed as leaning away
but also turning the whole body away. Space filing is the tendency to make oneself larger
or smaller, for example, by posture or more or less expansive gestures. Examples
postures include drawing up to full height or standing or sitting with legs apart. High
space filling is associated with dominance, low space filling with submission. Relaxation
is low bodily tension associated primarily with posture but also with other types of

behaviour. High relaxation is a sign of a dominant status, and can be expressed by asymmetry of posture.

In this embodiment of the invention, there are a number of high-level posture nodes that transform high level behavioural factors into actual postures. There is one high-level posture node for each posture. Each depends on one or more of the high-level behaviour types. The values of the high-level behaviour types are multiplied by a weighting to produce the value for a posture. This determines the degree to which the character is performing the posture. The weightings depend on the characters profiles, so that different characters would produce different postures for the same high-level behaviour. The values of the postures are then passed to the posture output node. This stores the actual representation of the postures. This is a representation in terms of joint angles. The joint angles corresponding to each posture are summed using the values of the postures as weights and the result is the actual posture of the character, which is passed directly to the underlying geometric representation.

Figure 8 shows certain stages in the generation of behaviour of a virtual object. The stages involved are: firstly, the design of one or more behavioural adjectives; secondly, the design of a behavioural profile which comprises a plurality of behavioural adjectives; thirdly, the assignment of values to the behavioural adjectives in the profile by a user; and finally, the subsequent generation of behaviour by the behavioural controller. The adjective and profile design stages generally occur off-line, whereas the user input and generation of behaviour by the behavioural controller can occur dynamically in real-time whilst the user is on-line in the virtual environment.

In the context of the invention, an adjective comprises a set of one or more behavioural parameters. The selection of which behavioural parameters in the framework of the behavioural controller affect a behavioural characteristic is a relatively skilled task. By providing adjectives however, the amount of data and understanding required of a user of the behavioural design interface is reduced. The adjective design 70 and the selection of default values 71 to assign a behavioural parameter has already been described hereinabove, with reference to Figure 7.

PROFILE DESIGN

In contrast to the level of knowledge required to set up adjective design, the selection of which adjectives should be included in the behavioural profile of an object 72 is a less complex task compared to the level of knowledge required to construct an adjective. The design stage of a behavioural profile enables a user to select which behavioural characteristics are relevant to the behaviour of the object(s) the user is seeking to control using the behavioural design interface.

The behavioural profile therefore consists of one or more adjectives. An adjective may comprise a single global or uniquely assigned behavioural node parameter value, or a plurality of one or more of each type of parameter name-value types. In this way a user can set internal and/or global parameters for the behavioural controller. In one embodiment of the invention, the behavioural profile comprises two sections, both of which are described using parameter name-value pairs. The first section describes the overall personality of the object (the term personality is used here to represent the general disposition of the object). The second section comprises a set of attitudes name value. In one embodiment of the invention, each adjective in the "personality" section comprises global parameter(s), whereas each adjective in the "attitude" section comprises unique behavioural node parameter(s).

Attitudes comprise aspects of the virtual object's behaviour that vary based on which other virtual objects are being interacted with by the virtual object. For example, a virtual object might be more friendly with one character than another. An attitude consists of the name of a character (or a set of characters) and a set of parameter values that are only loaded when interacting with that character. In this context, an attitude is a form of "adjective" in that it comprises a set consisting of at least one behavioural parameter name-value pair.

The attitude parameter section of the behavioural profile includes a set of at least one

parameter value for each named object present in an interaction. These values are
loaded into the parameters of the behavioural framework in order to generate appropriate
behaviour. A set of parameter values for a class of objects, or an individual object e.g., a
parameter value for objects of the class "stranger" can also be assigned to reflect the fact
that the object does not like other avatars which the avatar has not encountered before.

A parameter may have its value set in any appropriate manner. Two ways are
considered extremely appropriate. Firstly, a value can be directly specified by specifying
a frame work parameter using a node name, a parameter name, and a value to set the
parameter to. Secondly, a plurality of framework parameters may be associated in a
data structure already described herein called an "adjective", a term already defined
herein to refers to a set comprising one or more behavioural parameters of the
behavioural framework. A range of possible values a user may select for an adjective
may be included in the profile design stage 73 (or alternatively it may form part of the
adjective design stage).

Finally, once an "adjective" has been assigned a value by a user (step 74), the actual
values of each parameter in the set are determined in step 75 and are given by as a
function of the default values defined during the adjective design stage (step 73) and the
value assigned to the adjective by a user (step 74).
For example, a user may assign a value of '10' to the behavioural parameter set
"adjective" denoted "happy". When the behavioural profile is read into the behavioural
controller, the value '10' assigned by the user for "happy" is then translated into an actual
parameter value for all parameter(s) which have been determined by the adjective design
(steps 70,71) to make up the parameter set "happy", the actual value being determined
by a function operating on the input value which is associated with the adjective.

Which parameters collectively comprise a behavioural parameter set is determined by a
profile translation file. The profile translation file defines each behavioural parameter set
and associates with each an "adjective" name (or some equivalent a name or phrase for
the behavioural parameter set, i.e. a name intuitively associated with the behavioural
characteristic the parameter set modifies). The profile translation file also defines at

least one parameter as belonging to the behavioural parameter set. In summary, the final value in each parameter used by the behavioural model is the function of the value assigned in the profile and/or the value assigned to the behavioural parameter set.

There are several ways a user can modify a profile. For example, text-editing the code, assigning a value using a slider, or by direct manipulation of the virtual object, which is described in more detail later.

Returning now to Figure 8, the value assigned by the user to a behavioural adjective for the object is provided as input to the behavioural controller of the object. The behavioural controller then takes the input and infers from it which parameters should be assigned which values to produce suitable behavioural output (steps 75,76, 77, 78). The mechanism by which the input received is used to generate behaviour by the object is shown in more detail schematically in Figure 9A.

Figure 9A indicates the two ways in which the framework operates. Firstly, the framework can operate using forward mappings only, which occurs when a high-level input such as an external input or global parameter or adjective is set. In this case, the framework is run forward just generating outputs from inputs at the nodes (see also Figure 6) and doing forward mappings until the values of the output nodes are determined.

Secondly, the framework can operate to perform a reverse mapping when the outputs are changed rather than the inputs (again, see also Figure 6). For example, when direct manipulation of an object occurs. It is also possible where a plurality of inputs are received to perform both mappings, however, the reverse mapping can affect the output by the direct mapping and so in some embodiments of the invention this is performed first.

Referring now to Figure 9A, the behavioural controller receives input (step 81). If the input is received at an input node (step 81), for example, at a high level node in the framework corresponding to an externally input parameter or global parameter, then the input is then mapped forwards within the behavioural controller's framework of connected

nodes (step 83) to produce certain output (step 83) which is used to provide values to an animation system to generate the desired behaviour (step 85).   The desired behaviour comprises equivalent behaviour to the behaviour indicated at input.   However, the equivalent behaviour may be more complex and/or comprise more actions, some of which may be performed simultaneously and/or in a sequence.

REVERSE MAP

If instead, input is received from a source such as a direct manipulation of the object, then the input received is equivalent to the behaviour which the output one or more of the output nodes of the framework would produce (in conjunction with an animation system). In this case, the input is received at one or more of the output nodes (step 81) and is first reverse mapped through the behavioural framework to determine what input values would cause such output to be produced (step 82b).

This reverse mapping requires all relevant nodes in the framework to have their parameter values inferred, until the global parameters which would produce such behaviour are inferred (step 84,85).   These induced parameter values are retained for a forward mapping process starting from the inferred global parameter values and used to generate other behaviour.   The result is that although only one action was used to provide direct input, the behaviour produced by the controller can be much richer and complex than the original directly manipulated input, and can comprise one or more behavioural actions, or even a sequence of behavioural actions.   This is described again in more detail later.

FORWARD MAP

For example, consider an embodiment of the invention where the behavioural design interface assigns values to one or more global parameters, for example, global parameters representing certain moods of a virtual object such as an avatar, as well as perhaps certain node specific parameter values, for example, representing an attitude of the avatar towards another virtual object.   Referring now to both Figure 7 and Figure 9A, the input 62 is received by the appropriate nodes in the behavioural framework (step 82a) then mapped by the behavioural framework through internal behavioural nodes 63

(step 83) (for example, referring back to Figure 7, the parameter values may be mapped firstly through the immediacy equilibrium and dominance factor nodes, and then to the immediacy difference, responsiveness, and dominance difference nodes, which then map forward to the affiliation and status nodes, and the pleasantness, proximity, space filing and relaxation nodes, until reaching the output nodes 65 . In the embodiment of the invention shown in Figure 7, the output nodes comprise the facial expression, high level posture nodes (and this further maps to posture nodes), eye gaze, and gesture nodes, each of which generates output which can be provided to a suitable animation system to cause the avatar to be animated and generate the appropriate behaviour.   When the input is forward mapped through the framework, the global parameter are provided as input to other nodes in the framework, which enables secondary behavioural changes to be induced. These global parameter values thus enables more complex behaviour to be performed by the avatar.

DIRECT MANIPULATION

In the embodiment of the invention shown in Figure 7, the output nodes comprise the facial expression, high level posture nodes (and this further maps to posture nodes), eye gaze, and gesture nodes, each of which generates output which can be provided to a suitable animation system to cause the avatar to be animated and generate the appropriate behaviour.   Referring now also to Figure 9, consider an example where a user directly manipulates the posture of an avatar (e.g., by clicking on the avatar's arm step 90) for example, to touch another avatar's arm (step 91). This generates certain input at the posture node of the framework (step 92) The global parameter values which would produce such a behaviour by the avatar may be set up to do so only when friendliness towards the other avatar is intended. The behavioural controller performs a reverse map from the values produced by the posture node (step 93) back through the relevant nodes of the framework until reaching the global parameter input nodes of the framework, where an appropriate input value for the global behavioural parameter "friendliness" can be inferred (see step 93). These global parameter values are then used to start a forwards map through the framework.

The reverse map will have also assigned values to other behavioural parameter values in

38

the framework, for example, to pleasantness and relaxation (see Figure7). Running a forward map (step 94) from the inferred global parameter values, the values assigned are used to generate more output (step 95) which generates other behaviour (step 96). For example, the values assigned to pleasantness and relaxation, can generate additional behaviour at the output node for facial expression, which results in the avatar smiling. Other behaviour such as changing the eye gaze to look at the other avatar's face may also be produced, and a gesture such as handshaking may be generated to follow touching the arm. Thus although the user has only directly manipulated the avatar's posture to touch the arm of another avatar, the behavioural controller has interpreted the body language the user has given the avatar to infer that the avatar is friendly towards the other avatar and wishes to greet the avatar, and accordingly generates appropriate behaviour.

In summary, by inferring more information from the information received as input, the behaviour generated using a reverse map can comprise one or more behavioural actions which can be performed as a sequence, or at random, and one or more behavioural actions such as eye gaze, stance, motion etc., can be performed simultaneously. This greatly increases the complexity of the behaviour shown by the object, whilst also providing a very simple control mechanism for the user. The inference mechanism is described in more detail later.

Direct manipulation can be achieved by the user directly manipulating the virtual object using a mouse to click on a body part of the virtual object and then drag to body part into a new position. Other computer navigation tools or combination of tools, e.g. a cursor and the keyboard, a joystick, a track-ball, a pointer etc, can be used in any appropriate manner as is apparent to those skilled in the art to manipulate a virtual object.

In embodiments of the invention in which a real object is controlled by the behavioural controller, the user may wish to directly change the posture of the object by hand. The characteristics of this motion, which may include displacement and or the speed of the movement, as well as its proximity to other objects etc during the motion, will determine what input is fed to the behavioural controller.

In this way, a user is able to indicate a specific desired action by a virtual object, and the behavioural framework is then able to extrapolate more behavioural actions which are consistent with the behaviour the user has indicated is desired. The new output parameter values can be used to reproduce not only the action indicated by the user but also similar motion that is appropriate to different circumstances. Moreover, the parameter settings can then be either saved to a behavioural profile or as an adjective that can then be used by a user to build a behavioural profile. The direct manipulation of an object to trigger the generation of more complex behaviour is particularly useful where the object is a toy, as a child can then easily program the toy to perform complex behaviour.

The manner in which the behavioural framework for the behavioural controller of the virtual object infers the internal and global parameters is described is now described in more detail.

THE BEHAVIOURAL INFERENCE SCHEME

The structure of the behavioural framework defining the operation of the behavioural controllers enables internal parameter values to be inferred from input into the framework using a reverse map, for example when a user directly manipulates the virtual object. Referring back again to Figure 9A, consider the case where the behavioural controller receives input derived from a source external to the framework. Where the input is directly associated with output, e.g., if derived by direct manipulation, a reverse map needs to be performed to seed the global parameter values for the forward map. This reverse map is performed using the behavioural inference scheme outlined below.

An external source of input could be determined by the environment of the object. For example, the input may comprise information that another object has been thrown towards the object. Alternatively, it may comprise output from the behavioural controller of another object. Alternatively, it may comprise input from a user, for example, data provided by motion sensors attached to the user. Alternatively, it may comprise directly inputted values from a user, or input via the behavioural design user interface. The

40

input can provide an indication of the body language of another object, in which case this information can be used to infer an appropriate response.

Once input has been received by the behavioural controller (step 80), the received input values are then associated with output values for the object (step 81). This is performed using a special purpose map for each different type of behavioural output node. Once this initial mapping has been performed at the output node a reverse map is performed through the entire framework in order to infer internal parameters of nodes and global parameters. This is achieved using an equation set up for each affected output of each node in the behavioural framework:

$$T_{ij} = \prod_k F_{ijk}$$

where each factor is either an internal parameter or an input of the node.

In this way, each term Tij of the output has a solvable factor which distinguishes it from other terms Tij in the output. The terms Tij include solvable factors which may be internal values such as an internal parameter value or global parameter value. In such circumstances, the solvable factor is simply reassigned a new value. If the solvable factor is input from another node, then the process is iterated by forming a new set of equations to represent input from the source node. The change at the output is thus propagated up the hierarchy of nodes until the solvable factor can be represented by an internal or global parameter.

Each term Tij has two solvable parameters: the first is used for solving when inferring an internal state for real time control and the other is used for profile editing.

The output of each node in terms of a solvable factor is:

$$O_i = \sum_j f_{ij} S_{ij}$$

where fij is the sum of non-solvable factors and Sij is the solvable factor. This linear equation is then solved using suitable linear algebra methods. For example, where a

user has performed a number of editing functions, a sequence of linear equations exist in matrix form:

o = Fs

where **o** is a vector of the outputs of each of the edits, **F** is a matrix of non-solvable factors (which might depend on context, e.g. different settings of external inputs or time varying parameters, and **s** is the vector of solvable factors. As there will be more than one exact solution in all cases where F is not square, a pseudo-inverse method can be used to find a least-squares solution:-

s = F⁺o

This method enables the parameters of a single node to be inferred. In order to infer parameters for nodes further up the hierarchy which are not directly connected to outputs, some of the solvable factors are provided as inputs into a node rather than internal parameters The inputs can either be external inputs, global parameters or the outputs of other nodes. External inputs are not able to be solvable parameters. Global parameters have their values set directly during the solution process.

If the input is the output of another node, the solution requires solving for the other node, which can be achieved by setting up a system of equations including that node and any nodes which provide input into that node. In this case, the additional nodes provide output to other nodes, and so the equation to be solved is best expressed in the form

$$0 = \sum_j f_{ij} S_{ij} - O_i$$

each behavioural node to be determined from a given output node's output.

REAL TIME CONTROL

In embodiments of the invention where the final layer in the behavioural hierarchy enables a user to provide an object with complex behaviour in real time, the user may wish to provide input to the behavioural controller from a variety of sources or to use more than one behavioural design interface for any individual object. For example, or other devices which provide input may include using motion-trackers, for example on a

user's head and/or body. Two or more control devices may be operated simultaneously by a user to reduce the operational burden. The control device used exploits the use of adjectives to set certain parameters in the behavioural model to ensure that a user does not need to constantly control all the parts of the virtual object's body but only to manipulate a limited number of parts from time to time.

The behaviour of the object, whether derived by direct manipulation or sensor tracking a user generates information which can be passed to the behavioural controller using an appropriate interface mechanism. When the information represents parameter values which are equivalent to the parameter values of certain behavioural output nodes, the behavioural controller performs a reverse map through the framework to assign values internally to the relevant parameters needed to produce that particular behaviour by the virtual object. This enables an object to generate other behaviours associated with these parameters. For example, the user might animate an object's arm to touch another character's arm such as is shown in Figure 2C. The framework would infer that this gesture corresponds to a high value of flirtatiousness. This would then result in other flirtatious behaviour, for example, leaning forward and smiling.

The parameter values can represent a broad behaviour pattern, such as "friendliness", which may correspond to a user's behaviour in the case where trackers are used. No probabilistic mechanisms are used to infer the user's behaviour in this embodiment of the invention as the inference mechanism simply solves the algebraic equation relevant to the manipulation of the virtual object to determine what type of behaviour is being represented. Once recognised as "friendly" behaviour, the framework reproduces a wider range of friendly behaviour. This wider range of behaviour extends beyond what a user could be expected to directly control in real time, for example, a virtual object may be induced to smile by the framework, despite the fact that the user has only indicated the avatar is friendly by manipulating the avatar to give a handshake. If no input is provided, the framework enables the virtual object to continue to act autonomously. This enables a user unfamiliar with the behavioural design user interface to become more accustomed to the virtual environment and to learn about that environment without having to attempt to manipulate the virtual object right from the start. This provides a highly easy-to-use interface, suitable for even very young children.

The behavioural control system as shown in the accompanying drawings is based on an embodiment of the invention supporting non-verbal behaviour. In the general context of the invention, however, behaviour is represented by physical actions and/or vocal actions each modified by certain parameters. For example, physical actions such as body language may be modified by the current internal state of an object, e.g., whether the object is afraid. Vocal actions can be modified by parameters such as, for example, pitch. In this context, the term "behaviour" can be defined as one or more actions generated as a controlled or uncontrolled (reflexive) response to certain stimuli. Behaviour includes long term characteristic traits (for example, timid or aggressive) as well as more transient, emotional traits (for example, happy or sad).

An example of a non-verbal behaviour embodiment of the invention models an interpersonal relationship using the concepts of intimacy and dominance-submission. In this embodiment, the behavioural control system produces non-verbal activities to indicate various degrees of certain attributes such as affiliation, potency/status, which

44

reflect the level of intimacy and dominance/submission between parties. Affiliation is represented by a suitable behavioural display, for example, by activities such as smiling, physical closeness and/or touching. Potency/status can be represented, for example by displayed relaxation or tension in posture. Other non-verbal activities that can be

5   represented include responsiveness, displayed by general physical activation. In this context, the parameter values shown in Figure 1B, are in this embodiment of the invention predefined emotional/intra-personal parameters such as "friendliness" and "otherLiking" corresponding to affiliation or "machismo" or "selfImportance" corresponding to dominance.

10

OTHER EMBODIMENTS OF THE INVENTION

The embodiments described herein above primarily describe a virtual object's behaviour being controlled. However, the concepts of behavioural control described herein readily

15   extend to the control of non-virtual, i.e., real or tangible objects, even when not explicitly described in the above description, and the invention according extends to the behaviour of both real and virtual objects as will be appreciated by those skilled in the art.

The above embodiments of the invention have had behavioural frameworks which are

20   essentially static. In a static framework, once a parameter value has been set by a user it remains set to that value so until reset. More autonomous animation can be produced by varying the parameters over time. In order to produce real animation, the framework is modified to include time varying parameters.

25   Different types of time varying parameters can be supported by a behavioural framework. For example, a randomly varying parameter could provide occasional posture shifts while a different type could produce more predictable animations. Such parameters would require more sophisticated techniques for performing a reverse map, and so are not contemplated in the best mode of the invention. By controlling the alteration of any

30   timings of parameter changes, however, more realistic posture shifts may be obtained.

Another animated embodiment of the invention requires the provision of a number of

45

profiles, which would be changed between or interpolated over time. This would alter a set of parameters in one go. One way of implementing such an embodiment of the invention is to provide profiles as key frames in an animation sense and interpolate between them. Another way of implementing more animation is to control the character

5   using a finite state machine system where each state has a profile attached. The states would change due to events in the world or internal factors of the character and when a new state is entered its profile is loaded.

Other embodiments of the invention have other types of output node. For example, in

10   one embodiment of the invention, it is possible to provide only a posture node as an output node. This output node provides a simple mapping from parameters to behaviour, a parameter exists for each posture and the values of these parameters provide an interpolation between the postures. This model could be applied to other types of behaviour, for example, pieces of motion could be interpolated in the same way. More

15   complex behaviours will, however, requires more complex mappings, which increases the complexity of the mappings backwards from behaviour to infer internal parameters.

There are various approaches to this problem. The first is to ignore it and have some output nodes that can produce behaviour forwards but not be solved for. This is a

20   feasible option as long as there are other types of behaviour that can be controlled. Some aspects can be ignored as they are too hard for a user to control, for example, eye gaze. Another approach is to provide a hand crafted mapping for each type of output node. This is time consuming. The final way would be to use some sort of learning scheme to learn a mapping, for example, Bayesian or neural networks.

25

One embodiment of the invention combines a finite state machine extension and a probabilistic Bayesian network extension. Each node in the network has a number of states in this embodiment, each state having different internal parameters settings and as such different mappings between inputs and outputs. In this embodiment, the

30   probability of a node being in a given state would depend on the states of the other nodes, and global parameters or the outputs of other nodes. The framework then has two different types of interrelations between nodes. One is continuous as in the current

46

system with continuous valued parameters going between nodes. The other is discrete with nodes being one of a set of discrete states. Both types of information are given equal importance, as some behavioural features are naturally discrete, whereas other behaviour is not. For example, one can either cross one's arms or not, and one is either
5   in a conversation or not, 50% of either case being not very meaningful. However, other behavioural features are naturally continuous, for example, an object should be able to vary continuously between being happy and being angry over time and not do so as a sequence of discrete steps.

10   Different methods are suitable for inferring each type of information in such embodiments of the invention. The framework presented here is good for continuous data without uncertainty, whereas Bayesian networks can infer the probabilities of a set of discrete states. In some embodiments of the invention, an extension to the behavioural framework enables both methods for inferring information to be combined.

15

The method described hereinabove with reference to the accompanying drawings is non-probabilistic and can be used to infer the continuous state when the state of each node is known. In other embodiments of the invention where both inference methods are combined, then a Bayesian network can be used beforehand and node states and their
20   probability relationships between them can be treated as a Bayesian network and used to infer the states of each node from the output. The topology of the network would remain the same whether it is treated as a Bayesian or continuous network. Once these elements are in place the internal parameters of the system can be learned automatically from data of peoples behaviour.

25

In other embodiments of the invention, the methods described above for designing profiles can be extended to large numbers of examples to learn a set of internal parameters that are based on real data. Bayesian learning techniques can be used to learn probability relationships. With this the parameters of the system can be based on
30   data from real people and so can more accurately reflect people's behaviour which is very hard to capture theoretically.

47

An embodiment of the invention in which users interact via an on-line chat room is now described. In this embodiment, users are allowed to express themselves with body language as well as text providing input to the behavioural controller. This is also suitable for embodiments of the invention such as on-line counselling where an avatar represents

5   a human counsellor, as well as other virtual environments such as on-line mediation, on-line meetings (i.e., where information is to be exchanged by a group of virtual objects), on-line collaborations (i.e., where a task is to be performed by a group of virtual objects), and on-line commerce environments (e.g. avatar sales rooms).

10   The body language comprises behaviour generated autonomously by an architecture designed within the framework described, by using a set of predefined emotional/intra-personal parameters from a profile to generate the behaviour. The user accesses the system using a standard desktop PC. Computer navigation devices used by the user to provide input to the behavioural controller are limited to those commonly available in

15   such an environment, for example, a mouse and keyboard. The keyboard is used to input text and the mouse can be used to control the characters motion. As the user will mostly want to concentrate on typing text the control of the characters movement will be occasional. One input (e.g. via the behavioural design user interface) will result in a correction to the characters overall behaviour rather than a single motion, thus making

20   most use of a small amount of user input. This is easily achieved by inferring the emotional/intra-personal parameters from that input. More direct input of the avatar's emotions is done through the "emoticons" used in current graphical chat systems. These can directly set global parameters.

25   Another embodiment of the invention extends the above embodiment by further including input provided by the motion of the user. For example, home computer often have cheap video camera's, which can be arranged to perform face tracking through a specialised output node that provides input to the behavioural controller, the input being used to infer the user's state .

30

Embodiments of the invention described hereinabove allow a user to customise the behaviour of their avatar, by allowing them to edit their avatar's profile. For the end user

. 48

this would have to be done in as user friendly a way as possible, and the embodiments shown in Figures 1A to 2E is particularly suited to editing adjectives using the set of sliders provided by the behavioural design user interface.

5   The invention has many applications, including international avatar conferencing, which can be implemented in a manner similar to that for a chat room. In this case it is more important to capture the actual body language of the participants, particularly for business negotiations. This could be done by having more sophisticated body-tracking systems, including high quality face tracking the user. Other cues such as tone of voice

10   could also be used to infer the state of the user. The framework of the behavioural controller can be made more complex to ensure each user's behaviour is correctly inferred.

      In embodiments of the invention arranged for use in a computer game environment,

15   expressive body language can be generated and also inferred from the users commands for their character, what the internal state of their character is. The user input could be done a number of ways, for example, as described hereinabove, the user could directly manipulate the posture of the character. Alternatively, larger scale behaviour can be used for inference, for example, choice of action, whether to talk to some one or the

20   words chosen. Once the global parameters for the behavioural framework have been inferred, the characters in the game can be made to react to these. Advantageously, the invention provides a behavioural design tool which enables game designers to have fine control over the design of the behaviour of the characters using powerful yet intuitive tools.

25

      In embodiments of the invention where the object needs to perform a role, the profile a user creates using the design interface can be used to ensure behaviour is appropriate to the role and that any inference of internal state reflects the role of the object whose behaviour is inferred.

30

ROBOTIC APPLICATIONS

49

Whilst the embodiments described above are particularly relevant for virtual environments, the concept of providing a user with a simple input device to interface with a behavioural controller capable of generating complex, on-going behaviour has applications in the real world, in particular, for example, with robotic toys. Accordingly,

5   other embodiments of the invention provide a simple mechanism for a user to generate complex behaviour in a robotic object, particularly an articulate robotic object, such as a toy doll.

Such embodiments can enable a child, for example, to provide a robotic pet or toy with a

10   sophisticated character whilst requiring only manipulation of the intuitive labels assigned to the behavioural profiles. In such applications, the behavioural design user interface may be provided as a remote control type device. Behavioural control information can then be provided wirelessly to instruct appropriate action by the robotic device. The behavioural controller may be provided either as part of the interface device, with

15   animation instructions only transmitted to the robotic device, or as part of the robotic device itself, or as part of a proxy device which then relays behavioural instructions on to the robotic device.

In other embodiments of the invention, robotic devices used in manufacturing or

20   production line contexts may similarly require their behaviour to be controlled using a hierarchical behavioural model framework such as is described herein, and the behavioural design user interface may present behavioural options which are more role specific depending on the task the robotic device is to perform. Such robotic devices may be controlled remotely either by wired or wireless connections depending on their

25   context. For example, in a production line, the robotic device may be controlled via wired communications links, whereas in an underwater environment, a sub-sea robot may require a wireless communications link and/or a wired communications link.

Advantageously, a simple to use behavioural design interface over complex robotic

30   behaviour is provided. This is particularly important in applications where the robotic object has to perform a time-critical task with some level of autonomous/semi-autonomous behaviour that requires real-time control by the user. A hierarchical

50

behavioural framework is used to generate and control autonomous and semi-autonomous behaviour in an articulate object. A behavioural controller is arranged to receive input associated with a behavioural action, to infer a plurality of behavioural parameter values using the framework, and to generate equivalent behaviour in the

5   articulate object using the parameter values when loaded in the behavioural controller to generate output corresponding to the equivalent behaviour. The equivalent behaviour may reproduce the inputted behavioural action, and /or comprise one or more other behavioural actions, which may be performed simultaneously or as part of a sequence of actions.

10

BEHAVIOURAL TRANSLATION

The inference scheme described herein above enables a model of the internal state of a virtual agent or avatar to be determined which extends the application of the behavioural

15   controller to include the ability to translate between differing cultural behaviours. For example, in the case where the avatar is functioning as a personal virtual agent for a user who is interacting with users from different cultural backgrounds in an on-line environment such as a web-meeting, the ability to directly represent the user's actual movements and emotions may not be desirable.

20

Such embodiments of the invention may be modified to include translating the behaviour by the behavioural controller at either the input or output stages. This could be achieved by generating the external outputs using a behavioural framework corresponding to one culture and recreating the behaviour from this external output using a different framework

25   corresponding to the other culture. Otherwise the two frameworks could be the same but could use different profiles corresponding to the different cultures.

This can also be achieved by including a translation element which enables input received by the behavioural controller associated with a behavioural action

30   corresponding to a first culture to be first translated into input associated with an equivalent behavioural action in a second, differing culture. Similarly, it is possible to implement a translation element which receives the output from the output behavioural

51

nodes of the behavioural controller corresponding to behaviour according to the second culture and translates this back into the first culture.

The translation element can be provided within the controller to implement the translation of body language from one culture to another. One mechanism by which this can be achieved is by high-level parameters representing the meaning of a piece of body language being used with different profiles to generate different body language in different cultural contexts. This would ensure that other observers would perceive the participant's avatar's body language in their own culture. Thus a certain behaviour would input into the system by an observer of culture A, and would be interpreted as a friendly behaviour by the profile corresponding to culture A. The friendliness parameter would then be passed to a machine set with a machine containing a profile corresponding to culture B. This machine would generate a different behaviour but one which would have the meaning of friendliness in culture B.

The above embodiments of the method and apparatus for generating behaviour in an object need not be implemented in all embodiments of the behavioural translator, providing equivalent autonomous means are provided to map a behavioural display by an object which conforms to the one set of criteria to a behavioural display which conforms to another set of criteria. The mapping is implemented either by the system generating the behaviour and/or by the system displaying the behaviour to an observer.

In an embodiment where mapping is implemented by a behavioural translator provided as part of the system generating the avatar behaviour, it is necessary for the behavioural system to generate behavioural actions consistent with an indicated demographic group. This can be implemented by representing a demographic group by a set of one or more constraints modifying the behaviour the system generates and/or by one or more rules which map behaviour generated for other demographic groups to that demographic group.

In an embodiment where the behavioural translator is provided on an observer's system, the translator receives input comprising the output of the behavioural system which is

52

generating the behaviour. The behavioural translator may then include a set of rules for translating input representing one or more behavioural actions by an object into behaviour (which may include the omission of any actions or the generation of one or more equivalent behavioural actions for the other demographic group) appropriate for the

5   demographic group designed on the observer's system to be performed by the object when displayed one the observer's system.

The behavioural translator can be implemented in any virtual environment in which it is possible to map behavioural actions and/or displayed characteristics of a virtual object,

10   for example, a virtual animated articulated object such as an avatar, which conform to one demographic group within a population to equivalent behavioural actions and/or displayed characteristics conforming to another demographic group.

In a virtual environment, the behaviour translator translates behaviour (in particular

15   expressive behaviour) between avatars by sending information about the high level meaning of the behaviour, for example, rather than the individual inputs associated with each parameter of the behavioural framework such as is described hereinabove. The "high level" meaning can be inferred from the behaviour generated prior to the virtual object displaying this behaviour and used to generate parameter values for the

20   behavioural framework, or the parameter values can be determined directly or indirectly from input provided via a user interface. In both cases, it is possible for behaviour to be generated locally conforming to one demographic group and remotely conforming to a different demographic group, as the two behavioural profiles used to create the behaviour will be different. Alternatively, the high level behaviour only can be inferred

25   and this may be sent to the remote location where it undergoes appropriate translation.

As the method of generating behaviour described herein above is able to deal with high level behavioural information, it is possible to translate between different modalities, i.e., to translate between differing types of behavioural action. Thus it is possible for an

30   English gesture to be translated into a tone of voice in Japan and vice versa. Another example is if a gesture such as placing the finger-tips together is used to indicate a "what are you talking about gesture", this behaviour could be instead vocalised in one

53

demographic group, differently gesticulated in another and completely ignored by another demographic group.

It is also possible to translate high level meanings of behaviour, with some vocalised
5   explanation. For example, in certain demographic groups, if two avatars were present in an on-line meeting and one displayed extremely differential behaviour to the other, this could be occurring because the latter avatar is the boss/immediate superior of the first. Such differential behaviour may not be performed in other demographic groups. To explain to an observer from a latter demographic group the status of two avatars, it may
10   be helpful if the first avatar vocalised their relationship to the other avatar. There would be no need to provide this additional information for observers who share that same behavioural action in their own demographic group.

An embodiment of a behavioural translator according to the invention can be
15   implemented in a virtual environment which is implemented using a distributed computing environment. The distributed computing environment enables a plurality of users to participate in the virtual world. Each user is able to configure an application which enables them to participate in the virtual world to enable the user to perceive the behaviour of animated objects (e.g. avatars) in the virtual world as conforming with a
20   designated demographic group assigned by the user. The behavioural translation occurs locally (i.e., on the platform supporting the user's application) in one embodiment of the invention. Alternatively, the behavioural translation can be implemented remotely (e.g., on the platform supporting the application of the user whose avatar's behaviour is being translated).

25
One embodiment of the invention provides a behavioural translator suitable for enabling each observer of a virtual environment in which a plurality of avatars are interacting to perceive different behaviour according to different demographic group. The demographic group to which perceived behaviour is generated may be determined by the
30   behavioural controller of an avatar according to the demographic group designated by a user and which is represented by the avatar or according to the demographic group designated by an observer of the avatar in virtual environment.

54

Thus where a plurality of persons are participating in a virtual environment and each
person is associated with one or more avatars, it is possible for each person to associate
their avatar with a particular demographic group. This can be added to the profile for the
5   behavioural controller for an avatar. Alternatively, each observer can ensure that the
representation of the avatar which they are able to observer on the platform supporting
the application which enables the observer to observe and/or participate in the virtual
world translates certain behavioural actions into equivalent behavioural actions which
conform to the demographic group of the observer, regardless of whether a demographic
10   group has been designated in the behavioural system generating the avatar's behaviour.

Figures 10 A,B,C will be now described with reference to an embodiment of the
invention. Consider where the translation is implemented such that an avatar controlled
by a user A (not shown) participating in a virtual conferencing environment is intended to
15   display friendly greeting-like behaviour. In Figure 10A, a user has designated that
behaviour should be displayed according to an English demographic group. The high-
level "greeting-behaviour" output of the behavioural controller is then modified to
generate the behavioural actions comprising an avatar raising its hat. These behavioural
actions conform to a rule set/profile included in the behavioural system generating the
20   behaviour of the avatar associated with the first demographic group. Another participant
in the virtual world may designate a different demographic group, for example,
"American" and so they would perceive the "greeting-behaviour" to comprise a hand
wave (see Figure 10B). Another user may indicate a different demographic group (for
example, Japanese) where a more formal greeting may be required to convey the friendly
25   sentiment. This user will observer the avatar to bow as a friendly greeting (shown
schematically in Figure 10C). These different users may all be participating in the same
meeting on-line, but each will simply perceive a friendly greeting to have occurred
regardless of their different cultures.

30   In the above embodiment, the behaviour generated for the avatar may simply indicate a
greeting action is to be performed, and the output of the behavioural generator is
conveyed to each application which enables a user to participate in the virtual

55

environment where it is locally translated according to the user's preference Each system enabling a user to participate in the virtual environmentis provided with a behavioural translator which moderates the received behavioural output provided by the behavioural controller and which locally displays an image reflecting a demographic

5   group designated by the user.   Alternatively, the user'ssystem could exchange information with the behavioural controller actually generating the behaviour of the avatar (which would be located on a remote system) regarding a demographic group which the user has designated.   This information could be used to modify the behaviour generated on the user's system.   In this instance, the input received by theuser's local system

10   would not need translation on the local system and wouldenable behaviour to be directly displayed which the user would perceive as appropriate for the demographic group the user has designated.

It is particularly helpful if appropriate greeting behaviour etc, is provided when a plurality

15   of avatars are provided for the purposes of avatar conferencing. Avatar conferencing is described in more detail in WO 9954822, and WO9930494, the contents of which are hereby incorporated by reference.

In an embodiment of the invention comprising a virtual conference, where aplurality of

20   avatars participate in a virtual world environment,it is useful if the context of the meeting can be used to modify the behaviour. For example, if participating in an on-line learning environment, it is helpful if an avatar representing a pupil raises their hand prior to speaking/when requesting to ask a question, and if an avatar performing a teaching role indicates that the avatar can speak by pointing to the avatar which has raised his hand.

25   Appropriate behaviour for a virtual teaching environment can be contrasted with behaviour for other virtual environments, for example, a formal business conferencing virtual environment where status related behaviourmay not be appropriate. In another embodiment of the invention, the translation may also be affected by the role context, such as for example, teacher/pupil within a virtual classroom and teacher/pupil

30   interactions in other virtual environments. Examples of other context dependent cultural/demographic behaviour which may be translated include the reaction of an adult avatar when a child avatar enters a part of the virtual environment (e.g., a room), in which

56

case different demographic groups may display different reactions.

Examples of actions which may require behavioural translation between different cultures include the OK hand sign ('O' gesture forefinger finger and thumb or similar meanings).
5   This means OK for the English, but for the French it means 'zero' or 'worthless' if the person is not smiling. For the Japanese it can mean 'money'. Another gesture is tapping the side of the nose with the forefinger: in UK this can indicate conspiracy or secrecy, whereas in Italy it can mean take care or danger. Another action which enhances avatar behaviour in one culture but which would not be understood in another culture is a chin
10   flick (where a hand is positioned under the avatar's chin and is then projected forward). This can be interpreted as insult in France, and in North Italy may mean a sentiment such as 'get lost' or 'you are annoying me'. In contrast, in certain parts of South Italy it is no longer rude and conveys merely the sentiment 'No' or 'I don't want'.

15   Thus the invention provides a means for translating behavioural actions (which term includes behavioural displays) whenever displayed behaviour may convey a different semantic to different demographic groups.   In particular, where the semantic changes so much that the sentiment conveyed by the behavioural action changes. For example, the nodding head in many demographic groups means yes. But in others it can mean no.
20   Moreover, the behavioural action may be modified according to demographic group, such as with a nod which could start with the head high or low. Translation may be from one behavioural to a plurality of behavioural actions or vice versa.

To reflect the various behavioural actions, the profiles submitted to the behavioural
25   framework described herein above may be modified to reflect appropriate demographic behavioural rules.   Examples of demographic groups comprise cultures identifiable by various characteristics (and which in appropriate embodiments of the invention could be represented in a behavioural profile), including, for example, political characteristics, racial characteristics, national characteristics, religious characteristics, and
30   characteristics of social class, role/occupation, background. The demographic group may be modified in a virtual environment to reflect the role of a character, e.g., if the character is a member of the "royalty" of a virtual world, this would indicate the social

57

class would be high. If a priest, this could also indicate a high social class in some demographic groups, but in others this could be a very low social class. Thus, depending on the demographic group, if a priest interacted with a member of "royalty" the autonomous/semi-autonomous behaviour generated for the priest character participating
5   in the interaction would be very different.

Behavioural actions which may be translated include actions such as posture, tone of voice, manner of speech, gestures, eye gaze, nodding, the proximity of two avatars to each other in the virtual environment when performing certain behavioural actions,
10   stance, actions reflecting behavioural etiquette (for example, greetings) actions reflecting social status in the virtual world, actions reflecting a role being performed in the virtual world. Examples of greeting type behaviour include removing items of clothing such as raising one's hat, bowing, or greeting another avatar in a seated, sitting or lying posture (which could also reflect the avatar's social status in the virtual world). Examples of
15   social status related behavioural actions include behaviour which depends on who else is present in the virtual world environment, for example, if an avatars boss is present, the avatar may be more subservient, or observer more formal salutations and in certain demographic groups more formal salutations are the norm in business interactions whereas in other demographic groups such formal behaviour is not usually performed.
20
Behavioural displays may include modifying the appearance of an avatar to include tribal allegiance such as may be presented by body art where this is representative of a cultural or demographic group, and the appearance of clothing, for example, a female avatar dressed in one form of dress may wish to be perceived in a different attire when
25   participating in an online meeting with colleagues from a different culture. The behavioural translator may also translate behaviour according to the represented sex of an avatar (male/female) etc., as well as according to the role the avatar is to perform (e.g., teacher, parent, child, boss, minion etc)

30   Where the behavioural translator is provided as a means to translate real object behaviour, an example of a real object is a robot, for example, a robot waiter and/or a robotic guide. For example, if a robotic guide is implemented to enhance a visitor's

58

experience of a museum or other site of interest, then it is advantageous if the behaviour of the robotic guide can be changed according to the demographic group of one or more persons with whom the robotic guide is interacting.

5   It will be apparent to those skilled in the art that the invention can be implemented by an appropriate combination of hardware and/or software, and the combination of hard and software is not intended to be limited by the specific partition described hereinabove. Moreover, it is possible for the invention to be implemented by a suite of one or more computer programs running on one or more devices. The devices may be distributed

10   across a communications network .

The text of the abstract repeated below is hereby incorporated into the description:
A behavioural translator arranged for use by a behavioural controller of a virtual object, the translator arranged to map information representing behaviour conforming to a first

15   demographic group to behaviour conforming to a second demographic group. The behavioural translator comprises means to receive input; means to process said input to associate the received input with behaviour by the object conforming to the first demographic group; and means to map the behaviour derived from the received input to output generating behaviour conforming to the second demographic group.